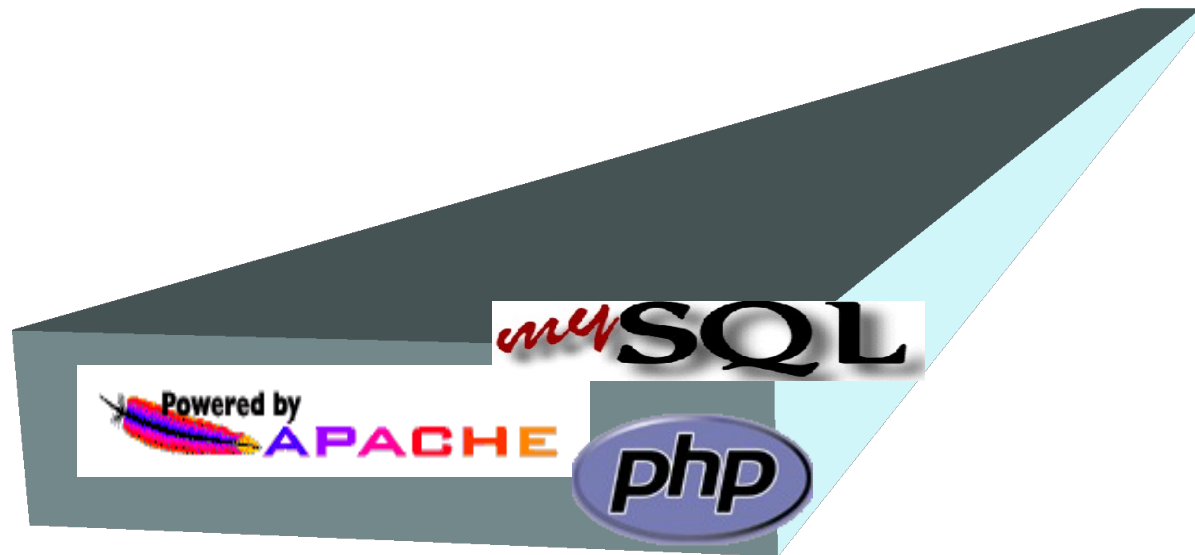




LAMP

Linux, Apache, MySQL, PHP

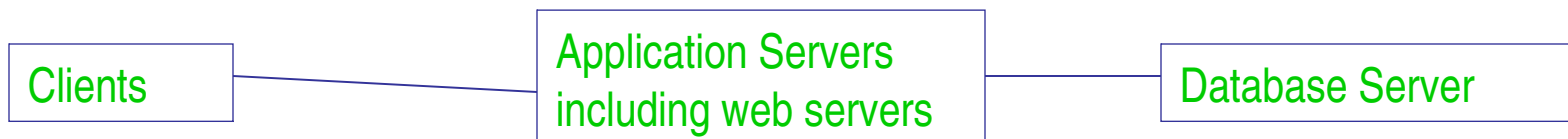
PHP in MySQL



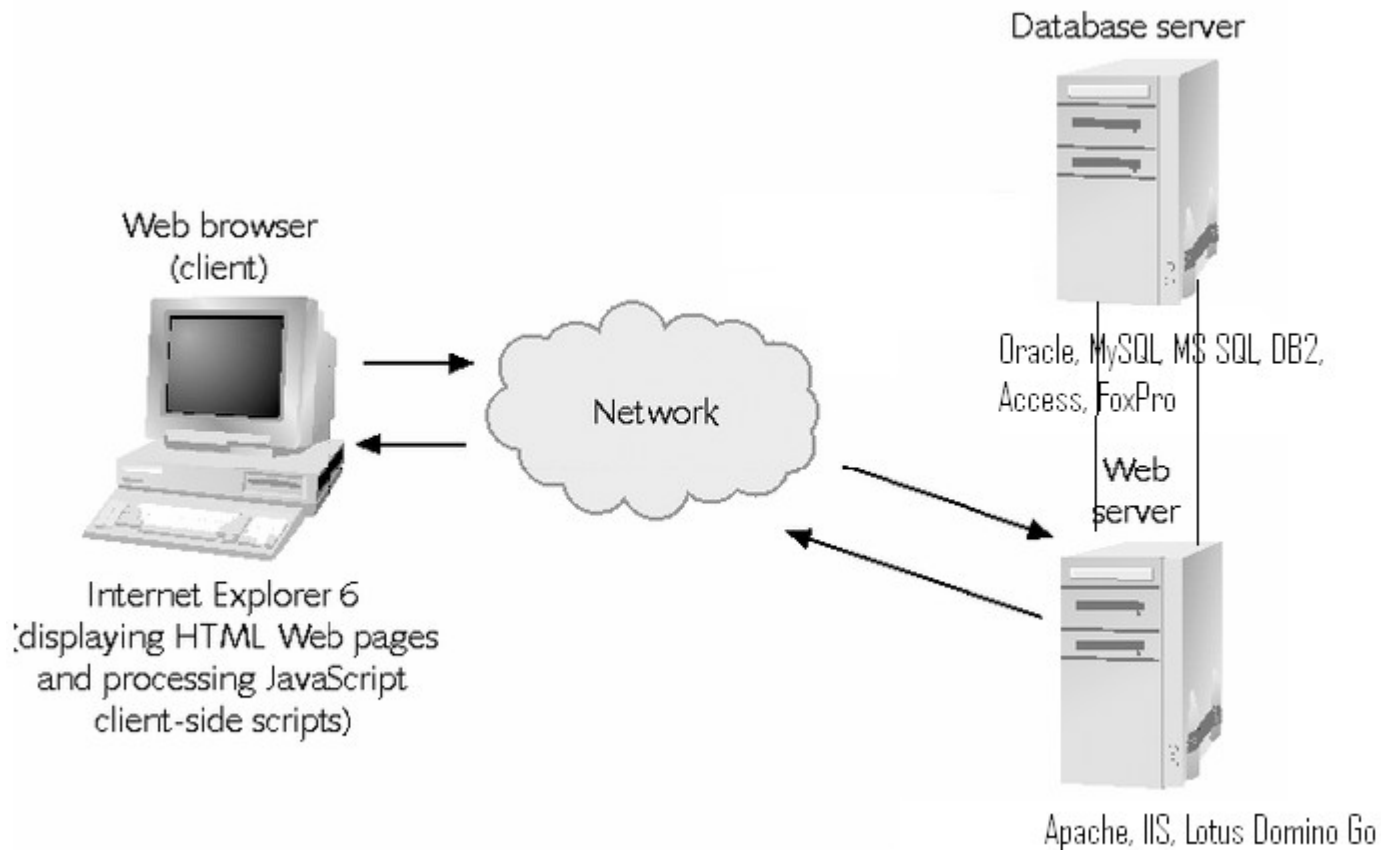
Spletne aplikacije s podatkovno bazo

database web application is an application based on World Wide Web (Internet) and database using web browser as a client.

- **Traditional client-server (2-tier architecture):** A client computer handles the user interface (Access Forms, Oracle Forms, Reports) and a database server stores the data. The actual functionality (business logic) of the application resides on the client and/or in the databases
- **Web-based (multi-tier architecture):** A client computer uses a browser to access information from two or more servers (web servers, application servers, database servers) (i.e.) A web server handles web requests, an application server handles dynamic requests and a database server stores the data)



Tro plastna arhitektura Three-Tier Architecture



Prednosti spletnih aplikacij



- **Free Infrastructure:** A major benefit is that the whole infrastructure is already in place and well developed. Only a web browser is needed. The new applications can be available to all the relevant users immediately – IT support staff do not have to go to each client workstation to install the Client Software
- **Free Upgrades:** As the application resides on the server, new versions will be immediately and simultaneously available to every user. There is no need to distribute updated application files to every user.
- **Interchangeable components:** It is possible to exchange either the server or the browser without breaking the application.

Spletne tehnologije – stran odjemalca

- HTML (HyperText Markup Language)
- CSS (Cascading Style Sheets)
- XML (Extensible Markup Language)
- DTD (Document Type Declaration)
- XSLT (Extensible Style Sheet Language Translator)
- JavaScript
- VBScript/Jscript
- Animation (Flash)
- Dynamic Hypertext Markup Language (DHTML – HTML, JavaScript, CSS & Document Object Model (DOM))

Spletne tehnologije – stran strežnika

- o PHP
- o ASP (Active Server Pages) & ASP.NET
- o JSP (Java Server Pages)
- o Java Servlets
- o C++/Java
- o PERL
- o XML

Podatkovne baze

- ODBC
- JDBC
- OleDb

LAMP



Linux (www.linux.com)

- *nix flavour that is all the rage at the moment.
- Similar to Unix but free and runs on just about anything.



Apache (www.apache.org)

- Most popular web page serving software

- Free
- Coordinated
- Cross-platform
- Plenty of support
- <http://www.lamphost.net/>

MySQL (www.mysql.com)

- Open source SQL database that is free and extremely powerful
-



PHP (www.php.net)

- PHP (Hypertext Preprocessor) is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more.



AMP

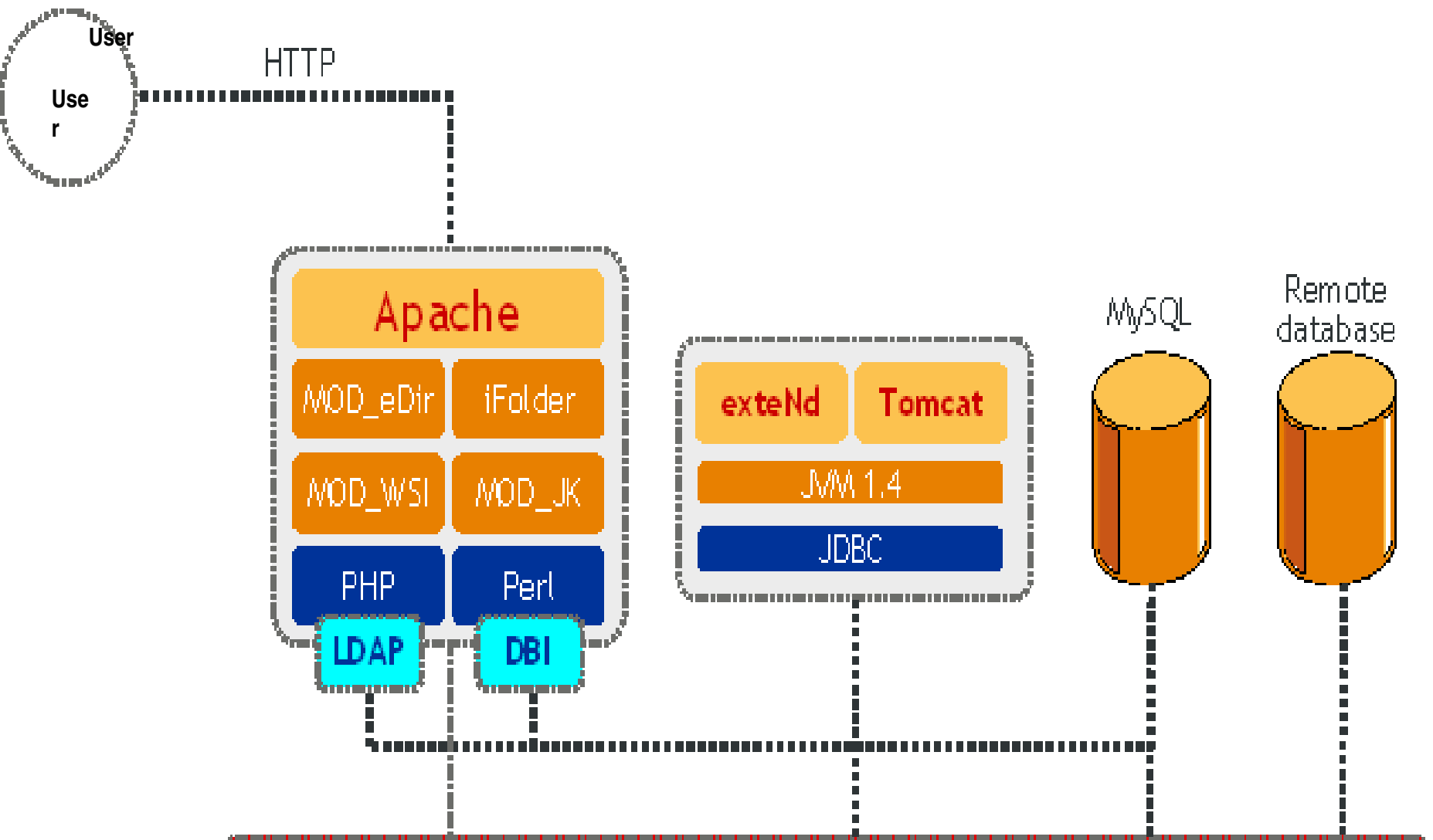
- One of the most powerful development models for the Web has been the notion of “AMP.”
- AMP stands for “**A**pache / **M**ySQL / **P**HP & Perl working together.”



Together, they form the nucleus of a web application system.

www.easyphp.org -> contains the software needed for all three packages. You can install and configure very easily. I am showing the installation separately

Integracija Apache, MySQL in PHP (AMP)



XAMPP



XAMPP

Many people know from their own experience that it's not easy to install an Apache web server and it gets harder if you want to add MySQL, PHP and Perl.

XAMPP is an easy to install Apache distribution containing MySQL, PHP and Perl. XAMPP is really very easy to install and to use - just download, extract and start.

Apache - Prednosti



- Apache is well **supported** - Most support for Apache is free and available 24 hours a day via Internet mail or newsgroups.
- Apache is **multi-platform** - Apache can run on virtually any hardware platform (from PCs to mainframes), and almost any operating system, such as Linux, Windows, NetWare, Macintosh, xBSD, etc.
- Apache is **secure** - security holes are rare but when they exist they are discovered and fixed quickly
- Apache is **extensible** - anyone can write modules that easily plug in to Apache. If Apache doesn't do what you want or need it to do, anyone with programming skills can write the modules you need.
- Apache is **database-friendly** - you can interface Apache with virtually any commercial database, such as Oracle, Sybase, DB2, and Informix, as well as free databases such as MySQL and Postgres.
- Apache is **hardware-friendly** - Apache generally consumes far fewer hardware resources than commercial web servers.
- **No Microsoft Viruses** - Apache is immune to the Code Red, Nimda, and other viruses that target at Microsoft Web servers.

MySQL - prednosti

- In 1996 T.c.X. DataKonsultAB , a consulting firm in Sweden developed MySQL.
- The largest growing relational database out on the market as it can handle large databases that can be accessed over the Web - Meets the ANSI SQL92 regulations (SQL-Structured Query Language)
- Mainly runs on UNIX-based environments, but also used on windows
- One of the most used open source databases in the world.
- Capacity to handle 50,000,000+ records.
- Very fast command execution, perhaps the fastest to be found on the market.
- Flexible and secure password system to protect your data - powerful security system
- Fast, reliable, easy to use, and affordable!
- On-line help facility - (*type -help or -?,*)
- Comes with a source code
- Multi-User and works on Several Platforms

PHP - Benefits



- Easy, powerful, popular
- **Server-side scripting** language
- Supports many DB's (not only MySQL)
- Platform Independent
- Web Server Independent
- Free and Open Source

PHP Overtakes Microsoft ASP as the Web's Number 1 server side Web technology for the Internet.

- An April Netcraft surveys indicate 24 percent of the 37.6 million websites, or are running PHP scripts. PHP adoption is growing by 6.5 percent each month. (9 million sites)

Test PHP and Apache

- Test the Apache Server as
- Create php folder in `D:\SoftwareInstallation\ApacheGroup\Apache2\htdocs` or where you have installed Apache
- Create `first.php` in `htdocs\php` folder (`D:\SoftwareInstallation\ApacheGroup\Apache2\htdocs\php`) with the following lines:

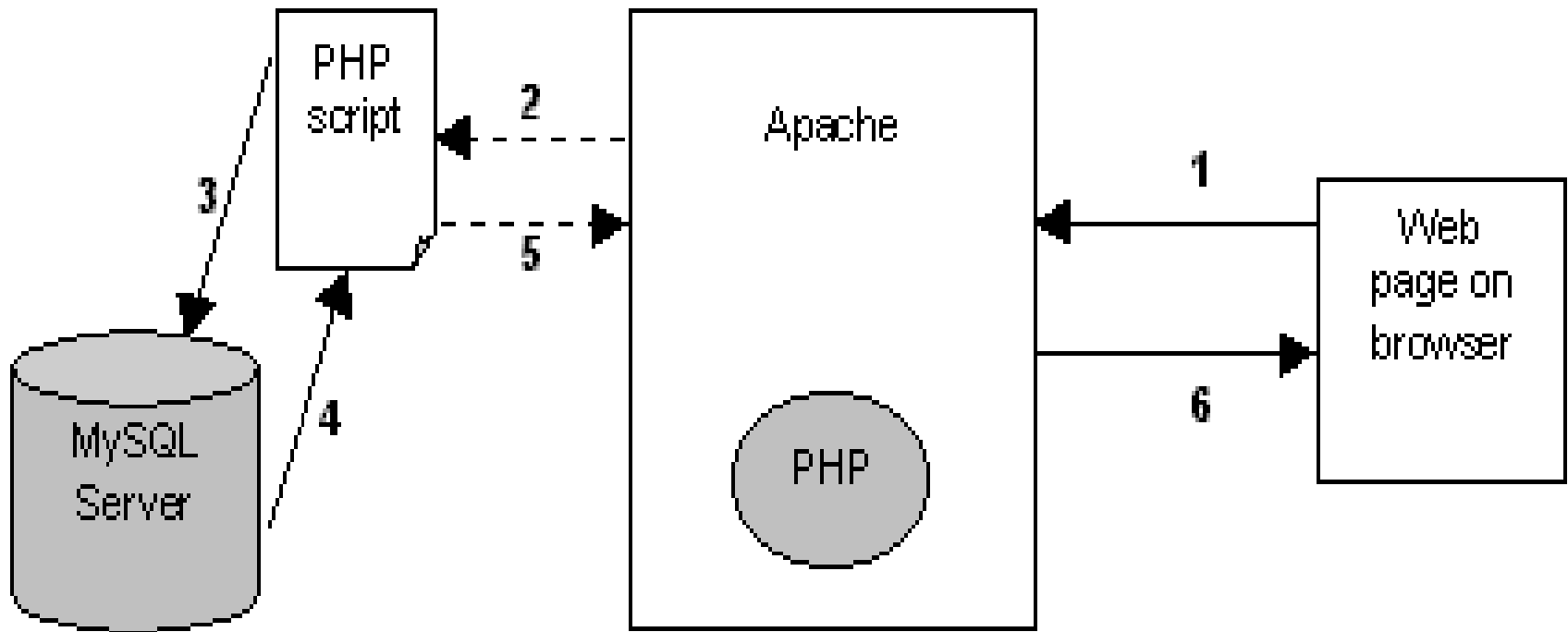
```
<?php  
    phpinfo();  
?>
```

- Open the browser and type the following link

<http://localhost:8088/php/first.php>

← the port number 8088 may be different in your installation. You will be prompted the php information on the browser.

How it works



MySQL Overview



- Fast, free, stable database
- Syntax is similar to Oracle
- Many of the same features as Oracle
- Production version still missing subqueries, stored procedures, and triggers
- Frequently used in conjunction with Linux, Apache, and PHP

MySQL – Database Basics

- A relational database manager (MySQL) manages databases which hold tables which have records (rows) with attributes (columns)
- Each record must have a unique ID, also known as a Primary Key. When used as an identifier in another table it's called a Foreign Key. Used for joins.
- Each attribute has to have a data type. (e.g. int, text, varchar)
- A database language (SQL) is used to create and delete databases and manage data

MySQL – Create Tables

- Table structure for following examples:

```
CREATE TABLE oscarpool (  
  uid int(4) auto_increment,  
  username varchar(255),  
  email varchar(255),  
  bestpicture int(2),  
  PRIMARY KEY (uid)  
)
```

```
CREATE TABLE  
bestdirector (  
  bdid int(4)  
  auto_increment,  
  name varchar(255),  
  PRIMARY KEY (bdid)  
)
```

- Created two tables, 'oscarpool' & 'bestdirector' using:
 - (a) use MySQL either in line mode (cd c:\mysql\bin and mysql) or
 - (b) Use MySQL Control Center
 - (c) use phpMyAdmin tool which you can get from
<http://www.phpmyadmin.net/>
- phpMyAdmin is a tool written in PHP intended to handle the administration of MySQL over the Web. Currently it can create and drop databases, create/drop/alter tables, delete/edit/add fields, execute any SQL statement, manage keys on fields, manage privileges, export data into various formats
<http://localhost:8088/mysql/index.php>

MySQL – INSERT

- Common SQL Statement: INSERT

```
INSERT INTO
```

```
oscarpool
```

```
(username,email,bestpicture)
```

```
VALUES
```

```
('dolsen','dave@usablecode.com',1)
```

- Creates a new record in the table 'oscarpool'
- Text fields need to have 's.
- Tip: If you have an ' in your data you need to escape it before inserting it. Can use the PHP function addslashes().
- Example: 'John O'Brien'

MySQL – SELECT

- Common SQL Statement: SELECT

SELECT uid,username

FROM oscarpool

- Selects the attributes 'uid' and 'username' from every record in 'oscarpool'
- SELECT is how you query the database. You can also:
 - limit the number of records returned with LIMIT,
 - limit retrieval to those records that match a condition with WHERE,
 - sort the data after the query has been evaluated using ORDER BY
- Tip: To easily select every attribute replace 'uid' with '*'

MySQL – UPDATE

- Common SQL Statement: UPDATE

```
UPDATE oscarpool
```

```
SET email = 'david.olsen@mail.wvu.edu'
```

```
WHERE uid = 1
```

- Updates the email address where 'uid = 1' in the table 'oscarpool'
- In this case I know that uid 1 is what my record was. In many cases you'd pass a uid variable from a form.

MySQL – DELETE

- Common SQL Statement: DELETE

```
DELETE FROM oscarpool  
WHERE uid = 1
```

- Deletes the record where 'uid = 1' in the table 'oscarpool'
- DELETE only removes the record from the table. To remove an entire table from the database you need to use the SQL statement DROP.
- Tip: To remove every record in a table but not remove the table just don't include the WHERE clause.

MySQL – JOIN



```
SELECT bd.name  
FROM oscarpool op, bestdirector bd  
WHERE op.uid = 1 and  
op.bestdirector = bd.bdid
```

- Selects the name of the Best Director that the user with 'uid = 1' has chosen
- bestdirector is a Foreign Key of the Primary Key for the table BestDirector

MySQL – DB Access

```
<html>
<body>
<h1>A List of Users Who Have Signed Up For OscarPool</h1>
<?
    $dbh = mysql_connect("localhost","root","")
        or die("Couldn't connect to database.");
    $db = mysql_select_db("test", $dbh)
        or die("Couldn't select database.");
    $sql = "SELECT username, email FROM oscarpool";
    $result = mysql_query($sql, $dbh)
        or die("Something is wrong with your SQL statement.");

    while ($row = mysql_fetch_array($result)) {
        $username = $row['username'];
        $email = $row['email'];
        echo '<a href="mailto:'. $email.'">'. $username.'</a><br />\n';
    }
?>
</body>
</html>
```

Save it as data.php

Povzetek

- Create MySQL database(s) with tables as required.
- Create PHP powered webpage for adding, deleting, updating and viewing database information.
- Be aware of security concerns vis-à-vis configuration file contents and database info.